

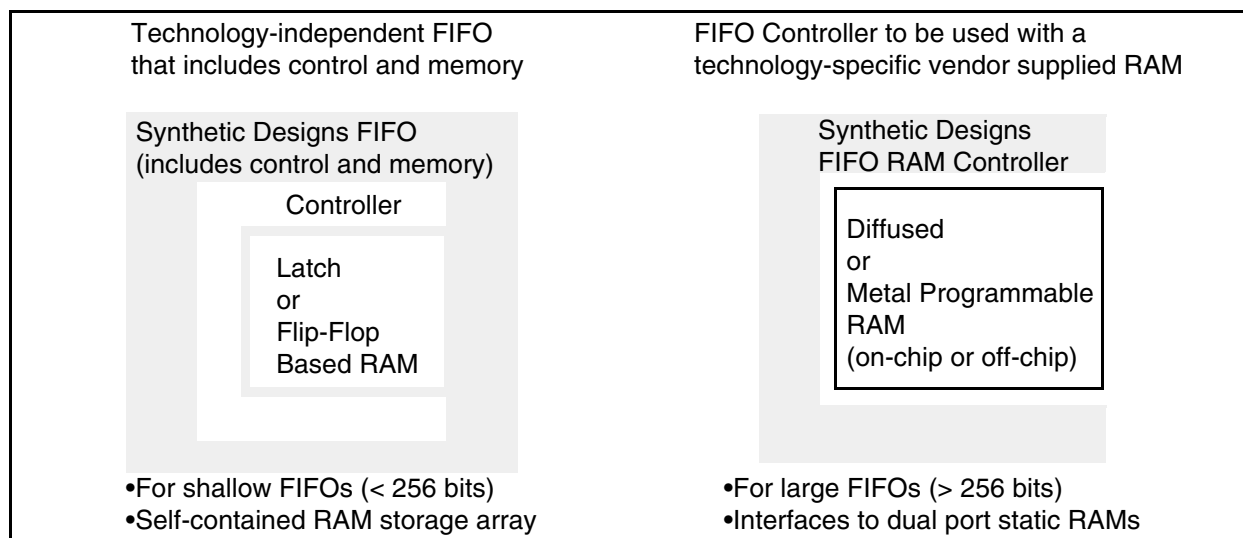
DWBB Memory – FIFO Overview

The FIFOs in this category address a broad array of design requirements. FIFOs, which include dual-port RAM memory arrays, are offered for both synchronous and asynchronous interfaces. The memory arrays are offered in two configurations: latch-based to minimize area, and D flip-flop-based to maximize testability. These two configurations also offer flexibility when working under design constraints, such as a requirement that no latches be employed. Flip-flop-based designs employ no clock gating to minimize skew and maximize performance. All FIFOs employ a FIFO RAM controller architecture in which there is no extended “fall-through” time required before reading contents just written.

Also offered are FIFO Controllers without the RAM array. They consist of control and flag logic and an interface to common ASIC dual port RAMs. Choosing between the two is typically based on the required size of the FIFO. For shallow FIFOs (less than 256 bits), synchronous or asynchronous FIFOs are available which include both memory and control in a single macro. These macros can be programmed via word width, depth, and level (almost-full flag) parameters.

For larger applications (greater than 256 bits), you can use the asynchronous FIFO Controller with a diffused or metal programmable RAM; see [Figure 1-1](#).

Figure 1-1 Memory: FIFOs and FIFO Controllers



All FIFOs and Controllers support full, empty, and programmable flag logic. Programmable flag logic may be statically or dynamically programmed. When statically programmed, the threshold comparison value is hardwired at synthesis compile time. When dynamically programmed, it may be changed during FIFO operation.

Related Topics

For a listing of Building Block components and associated datasheets, see:

- [DesignWare Building Block IP User Guide](#)

Copyright Notice and Proprietary Information

© 2023 Synopsys, Inc. All rights reserved. This Synopsys software and all associated documentation are proprietary to Synopsys, Inc. and may only be used pursuant to the terms and conditions of a written license agreement with Synopsys, Inc. All other use, reproduction, modification, or distribution of the Synopsys software or the associated documentation is strictly prohibited.

Destination Control Statement

All technical data contained in this publication is subject to the export control laws of the United States of America. Disclosure to nationals of other countries contrary to United States law is prohibited. It is the reader's responsibility to determine the applicable regulations and to comply with them.

Disclaimer

SYNOPSYS, INC., AND ITS LICENSORS MAKE NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.

Trademarks

Synopsys and certain Synopsys product names are trademarks of Synopsys, as set forth at <https://www.synopsys.com/company/legal/trademarks-brands.html>.

All other product or company names may be trademarks of their respective owners.

Free and Open-Source Software Licensing Notices

If applicable, Free and Open-Source Software (FOSS) licensing notices are available in the product installation.

Third-Party Links

Any links to third-party websites included in this document are for your convenience only. Synopsys does not endorse and is not responsible for such websites and their practices, including privacy practices, availability, and content.

Synopsys, Inc.
www.synopsys.com

The preceding non-breaking space is required so the text inset, when refreshed, does not cause an extra line that is forced to top-of-page. This is a work-around for text insets and described in Adobe forums.

